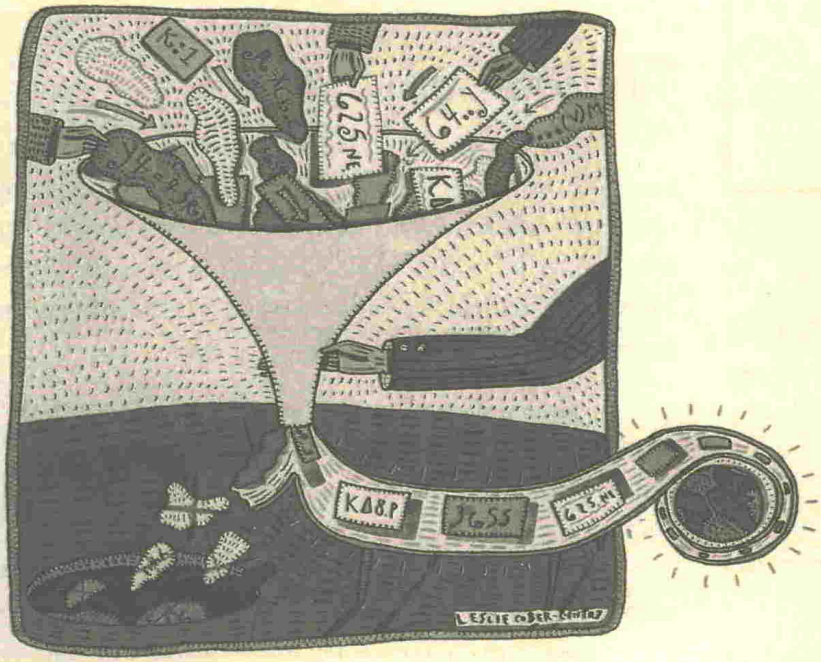
ซอฟต์แวร์
สุทธิศักดิ์ พงศ์ธนาพาณิช

Advanced BASIC Programming Technique

ตอน การสร้าง Fractal Sierpinski ด้วย VB/Win



เทคนิคการเพิ่มความเร็วของการแสดงผล fractal บนจอภาพ และเทคนิคการเพิ่มความเร็วในการพิมพ์ภาพอิมเมจออกทางเครื่องพิมพ์

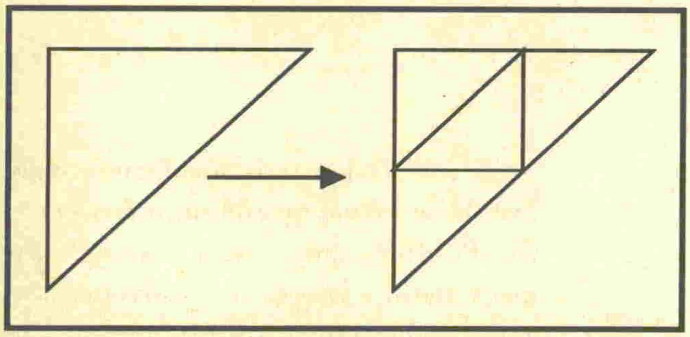
วิธีการสร้าง fractal ผมจะขอจบในตอนนี้ เพราะเท่าที่ผ่านมาผมคิดว่าผมได้เขียนเกี่ยวกับความรู้พื้นฐานและเทคนิคการเขียนโปรแกรมสร้าง fractal จากสมการคณิตศาสตร์เลขเชิงซ้อนด้วย VB/Win ไปบ้างแล้ว ซึ่งการศึกษาพื้นฐานของ fractal เราถือว่า Mandelbrot set และ Julia set เป็นสมการที่ใช้สร้าง fractal ที่คลาสสิคที่สุด และเป็นสมการที่มีชื่อเสียงมากทางด้านนี้

สำหรับในบทความตอนนี้ผมก็จะแสดงให้เห็นถึงการใช้สมการเชิงเส้นตรงมาใช้ในการสร้าง fractal กันบ้าง เพื่อไม่ให้ผู้อ่านคิดว่า fractal สามารถสร้างด้วยสมการ $Z(next) = Z(previous)^2 + c$ เพียงสมการเดียว แต่ในความเป็นจริงยังมีการนำฟังก์ชันอื่น ๆ อีกมากมายที่เกี่ยวข้องกับตัวเลขเชิงซ้อนได้ถูกนำมาใช้สร้าง fractal เนื้อหาในตอนนี้ผมจะไม่เน้นที่สมการของ sierpinski แต่จะใช้หน้ากระดาษที่เหลืออธิบายถึงเทคนิคการเพิ่มความเร็วของการแสดงผล fractal บนจอภาพ และเทคนิคการเพิ่มความเร็วในการพิมพ์ภาพอิมเมจออกทางเครื่องพิมพ์ ซึ่งทั้งสองเทคนิคผมได้ปรับปรุงจากตอนที่ผ่านมา โดยสาเหตุที่ผมไม่เสนอเทคนิคนี้ตั้งแต่ตอนแรก ๆ ก็เพราะต้องการให้ผู้อ่านเห็นว่าในการเขียนโปรแกรม เรามีวิธีที่สามารถเพิ่มประสิทธิภาพของโปรแกรมได้หลาย ๆ วิธี เพียงแต่ปรับเปลี่ยนขั้นตอนการทำงาน (อัลกอริธึม) เท่านั้น ก็สามารถทำให้โปรแกรมทำงานได้รวดเร็วขึ้น และถ้าหากผู้อ่านท่านใดเคยดูโปรแกรมตัวอย่างที่ติดมากับ VB/Win หรือหนังสือ Insider ของต่างประเทศ ก็จะเห็นว่าเทคนิคที่ผมนำมาใช้ในโปรแกรมตัวอย่างประกอบบทความตอนนี้ สามารถทำงานได้เร็วกว่าหลาย ๆ เท่าตัวเลยทีเดียว จะเร็วกว่ายังไงจริงหรือไม่ก็เชิญติดตามกันเลยครับ

ทฤษฎีของ Sierpinski

รูปทรง fractal นี้ได้ถูกค้นพบโดยนาย W. Sierpinski ในช่วงสงครามโลกครั้งที่ 1 ซึ่งแนวคิดในการสร้างรูปทรง sierpinski gasket ก็เริ่มต้นจากการแบ่งสามเหลี่ยมออกเป็น 4 ส่วน โดยอาศัยรูปสามเหลี่ยมที่เหมือนเดิมแต่มีขนาดเล็กกว่าเดิม โดยการวางตำแหน่งสามเหลี่ยมเล็กไว้ในสามเหลี่ยมใหญ่ และให้ปลาย

ทั้งสามของสามเหลี่ยมด้านในอยู่ที่จุดกึ่งกลางของเส้นขอบรูปสามเหลี่ยมด้านนอก ดังในรูปที่ 1 แสดงขั้นตอนการสร้าง Sierpinski gasket และจากขั้นตอนที่สองก็จะได้สามเหลี่ยมเพิ่มขึ้นเป็น 4 รูป หลังจากนั้นก็ให้แบ่งสามเหลี่ยมทั้ง 4 รูป ออกเป็นสามเหลี่ยมเล็ก ๆ อีกอย่างละ 4 รูป ทำเช่นนี้ไปเรื่อย ๆ ก็จะได้ fractal ทันที



รูปที่ 1 แสดงขั้นตอนการสร้าง sierpinski gasket

จากขั้นตอนการสร้างดังกล่าวจึงได้มีผู้คิดค้นสมการคณิตศาสตร์เชิงเส้นตรง เพื่อให้สามารถสร้างรูปทรง fractal ได้โดยทำการคำนวณแบบซ้ำซ้อน ซึ่งสมการดังกล่าวมีรายละเอียดดังตัวอย่างที่ 1

ตัวอย่างที่ 1

```
If Z(n).b > .5 Then
    Z(n+1).a = 2 * Z(n).a
    Z(n+1).b = 2 * Z(n).b - 1
ElseIf Z(n).a > .5 Then
    Z(n+1).a = 2 * Z(n).a - 1
    Z(n+1).b = 2 * Z(n).b
Else
    Z(n+1).a = 2 * Z(n).a
    Z(n+1).b = 2 * Z(n).b
End If
```

จากสมการข้างต้น ผมจึงได้นำมาเขียนเป็นโค้ดลงในโปรแกรมตัวอย่างได้ดังตัวอย่างที่ 2

ตัวอย่างที่ 2

```
If Z1.b > .5 Then
    Zn.a = 2 * Z1.a
    Zn.b = 2 * Z1.b - 1
ElseIf Z1.a > .5 Then
    Zn.a = 2 * Z1.a - 1
    Zn.b = 2 * Z1.b
Else
    Zn.a = 2 * Z1.a
    Zn.b = 2 * Z1.b
End If
```

ส่วนวิธีการคำนวณแบบซ้ำซ้อนและการตรวจสอบเงื่อนไขการออกนอกขอบเขตของ fractal ก็เหมือนกับวิธีที่ใช้ในการสร้าง mandelbrot set ขอให้อ่านรายละเอียดในฉบับที่ผ่าน ๆ มาครับ ส่วนวิธีการใช้งานโปรแกรมผมก็ขอกล่าวเพียงย่อ ๆ เพราะจะเหมือนกับโปรแกรมเกี่ยวกับ fractal ใน 2 ตอนที่ผ่าน ๆ มา แต่ในตอนนี้ผมจะขอเน้นที่การทำงานของรูทีนการวาดและพิมพ์ fractal ว่ามันมีวิธีเพิ่มความเร็วในการแสดงผลอย่างไร และจริงหรือเปล่าที่ผมเกริ่นไว้ในตอนต้นของบทความว่าเทคนิคของผมสามารถทำงานได้เร็วกว่าเทคนิคที่เขียนไว้ในหนังสือ Insider!

โปรแกรมตัวอย่าง SIER

โปรแกรมตัวอย่างสร้างรูปทรง fractal จากสมการเชิงซ้อน Sierpinski ผมได้ออกแบบฟอร์มให้มีหน้าตาดังในรูปที่ 2 ซึ่งผมได้ออกแบบฟอร์มให้คล้ายคลึงกับฟอร์มของโปรแกรม MANDEL ในฉบับที่ผ่าน ๆ มา เพื่อให้ผู้อ่านมีความคุ้นเคยและสามารถเรียนรู้การใช้งานโปรแกรมได้ง่าย โดยที่แต่ละ text box จะเป็นค่าที่ใช้กำหนดขอบเขตการทำงานของโปรแกรมสร้าง fractal โดยมีความหมายดังนี้

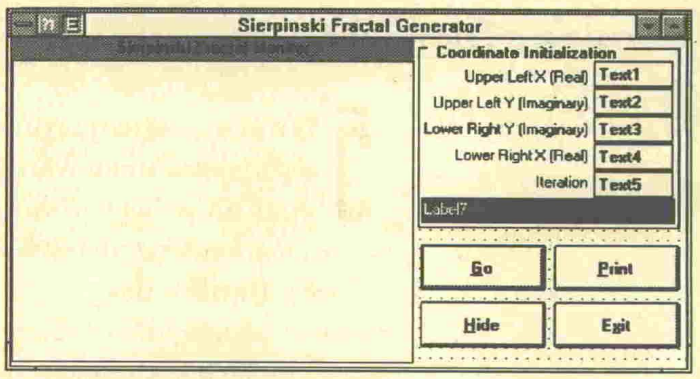
Upper Left X (Real) หมายถึง ค่าในแนวแกนนอนของขอบซ้ายสุด

Upper Left Y (Imaginary) หมายถึง ค่าในแนวแกนตั้งของขอบบนสุด

Lewor Right X (Real) หมายถึง ค่าในแนวแกนนอนของขอบขวาสุด

Lower Right Y (Imaginary) หมายถึง ค่าในแนวแกนตั้งของขอบล่างสุด

Iteration หมายถึง จำนวนครั้งในการทำ Iteration ของแต่ละตำแหน่งเลขเชิงซ้อน



รูปที่ 2 แสดงฟอร์ม Form1 (SIER.FRM) ในขณะออกแบบ

ส่วน Label7 ก็ถูกใช้ในการแสดงตำแหน่งของโคออร์ดิเนตเชิงซ้อนที่กำลังถูกคำนวณในการสร้าง fractal และปุ่มคำสั่งต่าง ๆ ในแนวด้านขวาล่างสุดของฟอร์มก็มีหน้าที่ดังนี้

Go ทำหน้าที่วาดรูปทรง fractal โดยใช้ค่ากำหนดขอบเขตตามที่ได้กำหนดเอาไว้

Hide ทำหน้าที่ซ่อนส่วนแสดงผลรูปทรงเอาไว้ในขณะกำลังประมวลผล fractal

Print ทำหน้าที่พิมพ์รูปทรง fractal ใน picture control ออกทางเครื่องพิมพ์

Exit ทำหน้าที่จบการทำงานของโปรแกรม

ตัวอย่างที่ 3

```
Left = (Screen.Width - Width) \ 2
Top = (Screen.Height - Height) \ 2

Text1.Text = "-0.15216"
Text2.Text = "1.1216"
Text3.Text = "1.10216"
Text4.Text = "-0.1216"
Text5.Text = "35"
Label7.Caption = "Status [Ready]"
```

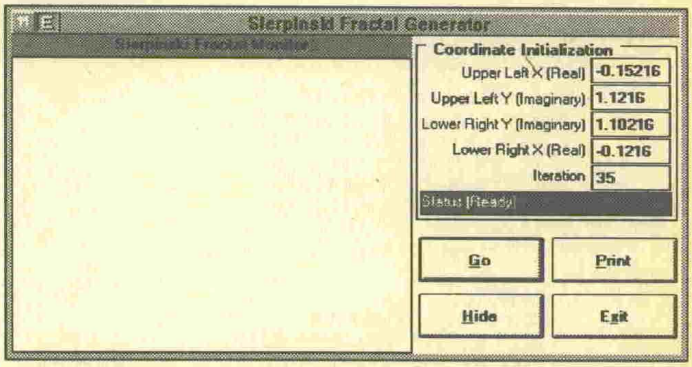
การทำงานของโปรแกรม SIER

เมื่อโปรแกรมถูกรันเหตุการณ์ Form_Load ก็จะถูกเรียก ถือได้ว่าเป็นขั้นตอนเริ่มต้นมาตรฐานของ VB/Win ซึ่งภายในก็จะมีการแสดงผลฟอร์มตรงกึ่งกลางของจอภาพและกำหนดค่าดีฟอลต์ให้กับพารามิเตอร์สำหรับการสร้าง fractal ด้วยโค้ดดังตัวอย่างที่ 3

ตัวอย่างที่ 4

```
If Command1.Caption = "&Go" Then
    TerminatedFlag% = False
    Command1.Caption = "&Cancel"
    CreatSierpinski
Else
    TerminatedFlag% = True
    Command1.Caption = "&Go"
End If
```

หลังจากนั้นฟอร์ม SIER.FRM ก็จะถูกแสดงดังในรูปที่ 3 ถ้าหากผู้อ่านไม่ต้องการแก้ไขค่ากำหนดขอบเขตสำหรับแสดงผล fractal ก็ให้เริ่มต้นวาดรูปทรง fractal ได้ทันที โดยการกดปุ่ม Go ซึ่งโปรแกรมก็จะกระโดดไปยังโค้ดในส่วนของเหตุการณ์ Command1_Click ส่วนรายละเอียดของโค้ดในส่วนนี้มีดังตัวอย่างที่ 4



รูปที่ 3 แสดงฟอร์ม Form1 (SIER.FRM) ในถูกโหลดโดยรูทีน Form_Load

ตัวอย่างที่ 5

```
    Picture1.Cls

    ' Find number of pixel in picture control.
    PixX% = Picture1.ScaleWidth - 1
    PixY% = Picture1.ScaleHeight - 1

    ' Calculate width and height of coordinate.
    RealRange = Abs(Val(Text3.Text) -
Val(Text1.Text))
    ImagRange = Abs(Val(Text4.Text) -
Val(Text2.Text))
    DeltaR = RealRange / (PixX% + 1)
    DeltaI = ImagRange / (PixY% + 1)

    IterNum% = Val(Text5.Text)
```

ถ้าหาก caption เป็น &Go ขบวนการสร้าง fractal ก็จะถูกเริ่มต้นทันทีโดยรูทีน CreatSierpinski ซึ่งโค้ดในรูทีนนี้จะเป็นส่วนที่ผมจะเน้นอธิบายการทำงาน โดยเริ่มต้นด้วยการคำนวณขนาดของ picture control ที่ใช้แสดงผล fractal เพื่อทำการคำนวณย้อนกลับเป็นตำแหน่งโคออร์ดิเนตเชิงซ้อนต่อไปดังตัวอย่างที่ 5

ต่อจากนั้นก็เป็นขั้นตอนการสร้าง fractal ซึ่งภายในรูทีนนี้ ผู้อ่านจะเห็นว่ามีส่วนแตกต่างจากรูทีนที่ใช้สร้าง fractal ในตอนที่ผ่าน ๆ มาเฉพาะในส่วนของการแสดงผลเท่านั้น โดยในตอนที่ผ่านมาผมใช้วิธี เมื่อตำแหน่งจุดใดถูกคำนวณก็จะแสดงผลตำแหน่งนั้น ๆ ลงในคอนโทรล picture box ทันที แต่สำหรับเทคนิคที่นำมาใช้ในบทความตอนนี้เป็นการแสดงผลตำแหน่งทีละคอลัมน์ โดยค่าสีที่จะแสดงผลในแต่ละตำแหน่งของคอลัมน์หนึ่ง ๆ จะถูกจัดเก็บไว้ในตัวแปรอะเรย์ ColorArr() แล้วจะนำตำแหน่งในคอลัมน์นั้นมาแสดงผลก็ต่อเมื่อทุก ๆ จุดในคอลัมน์นั้น ๆ ได้ถูกคำนวณไปหมดแล้ว ส่วนสาเหตุที่ผมใช้วิธีการแสดงผลเป็นกลุ่มและมันสามารถทำงานได้เร็วกว่าการแสดงผลทีละจุดเพราะ

ตัวอย่างที่ 6

```
    Z.a = Val(Text1.Text)
    ' Begin the main processing loop.
    For X% = 0 To PixX%
        If X% Mod \2 Then
            Label7.Caption = "Status
[Processing...]"
        Else
            Label7.Caption = "Status
[Processing   ...]"
        End If
        Z.a = Z.a + DeltaR
        Z.b = Val(Text2.Text)
        ReDim ColorArr(0 To PixY%) As Integer
        For Y% = 0 To PixY%
            DoEvents
            If TerminatedFlag% Then GoTo
SubTerminated
            Z.b = Z.b - DeltaI
            Z1 = Z
            InSetFlag% = True
            For i% = 1 To IterNum%
                If Z1.b > .5 Then
                    Zn.a = 2 * Z1.a
                    Zn.b = 2 * Z1.b - 1
                ElseIf Z1.a > .5 Then
                    Zn.a = 2 * Z1.a - 1
                    Zn.b = 2 * Z1.b
                Else
                    Zn.a = 2 * Z1.a
                    Zn.b = 2 * Z1.b
                End If
                If ModulusComplex(Zn) > 2 Then
                    InSetFlag% = False
                    Exit For
                End If
                Z1 = Zn
            Next i%
            ' save pixel color.
            If InSetFlag% Then ColorArr(Y%) =
0 Else ColorArr(Y%) = (i% Mod 15) + 1
        Next Y%
        ' show each column.
        For Y% = 0 To PixY%
            Picture1.PSet (X%, Y%),
QBColor(ColorArr(Y%))
        Next Y%
        ' Refresh the picture.
        Picture1.Refresh
    Next X%
```

อะไร ก็ขอให้ผู้อ่านกลับไปอ่านบทความเดียวกันในตอน DEVICE CONTEXT ก็จะทราบขั้นตอนการวาดจุดของ VB/Win ครับ สำหรับโค้ดส่วนที่ใช้เทคนิคช่วยเพิ่มความเร็วในการแสดงผลของโปรแกรมผมจะแสดงด้วยตัวเข้ม ดังตัวอย่างที่ 6

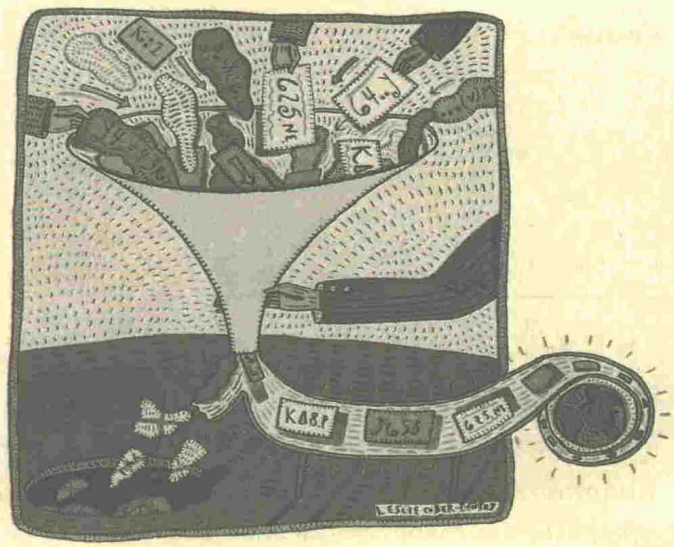
ส่วนรูทีนถัดไปที่มีความสำคัญก็คือ รูทีนการพิมพ์รูป fractal ออกทางเครื่องพิมพ์ โดยถ้าหากผู้อ่านทำการกดปุ่ม Print เหตุการณ์ Command3_Click ก็จะถูกเรียกทันที ซึ่งภายในรูทีนเหตุการณ์นี้ก็จะเป็นส่วนของโค้ดที่ทำหน้าที่พิมพ์รูปทรง fractal ออกทางเครื่องพิมพ์ ซึ่งเทคนิคที่ใช้ในการพิมพ์ (จริง ๆ แล้วเป็นการดัมพ์คอนโทรล picture box มากกว่า) เป็นการอ่านค่าสีจากแต่ละจุดใน picture control แล้วสั่งพิมพ์ออกทางเครื่องพิมพ์ ซึ่งขั้นตอนการพิมพ์ก็เริ่มต้นด้วยการหาขอบเขตของ picture control และค่าอัตราส่วน twip per pixel (TPP) ของจอภาพทั้งในแนวแกน X และ Y ซึ่งค่า TPP นี้ผมได้นำมาใช้ในการพิมพ์แต่ละจุดที่อ่านได้จากคอนโทรล

ตัวอย่างที่ 7

```
    ' Find number of pixel in picture control.
    PixX% = Picture1.ScaleWidth - 1
    PixY% = Picture1.ScaleHeight - 1
    ' Declare and initialize screen resolution
variables.
    ScrX% = Screen.TwipsPerPixelX
    ScrY% = Screen.TwipsPerPixelY
    ' For each row of pixels in the source
bitmap.
```

picture box ออกทางเครื่องพิมพ์ เพราะหน่วย TWIP จะเป็นหน่วยที่ละเอียดที่สุดในการแสดงผลบนวินโดวส์ ดังนั้นรูปที่พิมพ์ได้อาจจะมีขนาดไม่เท่ากับขนาดที่ปรากฏในคอนโทรล picture box ก็ได้ ทั้งนี้เนื่องจากวินโดวส์มีการแบ่งสเกลของจอภาพและเครื่องพิมพ์ไม่เท่ากันนั่นเอง แต่รูปที่พิมพ์ได้จะมีสเกลเดียวกัน (รูปที่พิมพ์ได้จะมีสัดส่วนที่ได้อัตราส่วนเดียวกัน) (ดูตัวอย่างที่ 7)

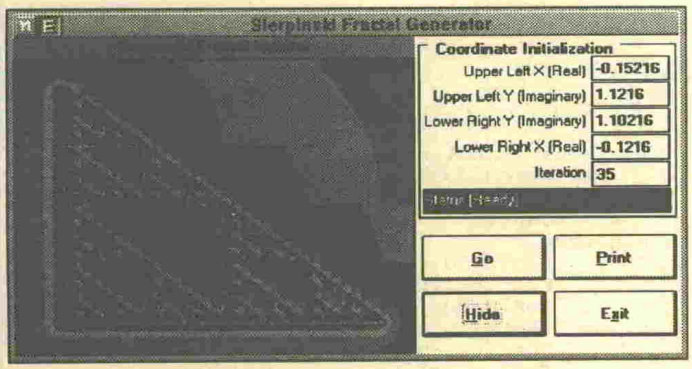
ส่วนเทคนิคในการแสดงผลรูปทรงออกทางเครื่องพิมพ์ก็คล้ายกับ เทคนิคการแสดงผลออกทางจอภาพ โดยการอ่านค่าสีแต่ละจุดของแถวนั้น ๆ และนำค่ามาเก็บไว้ในตัวแปร PelC() เพื่อที่จะนำค่าในตัวแปรนี้ส่งให้กับเครื่องพิมพ์ทีละแถว ซึ่งการแสดงผลโดย VB/Win ไม่ว่าจะบนจอภาพหรือเครื่องพิมพ์ ถ้าหากสามารถแสดงผลทีละกลุ่มได้ก็จะ



ตัวอย่างที่ 8

```
    For YRd% = 0 To PixY%
        If YRd% Mod 2 Then
            Label7.Caption = "Status
[Printing...]"
        Else
            Label7.Caption = "Status [Printing
...]"
        End If
        Label7.Refresh
        PelY% = Picture1.Top + (YRd% * ScrY%)
        ' For each pixel in the current row of
the source bitmap.
        ReDim PelC(0 To PixX%) As Long
        For XRd% = 0 To PixX%
            ' Store the pixel color in a local
variable.
            PelC(XRd%) = Picture1.Point(XRd%,
YRd%)
        Next XRd%
        For XRd% = 0 To PixX%
            PelX%=Picture1.Left+(XRd% * ScrX%)
            Printer.Line (PelX%,PelY%)-
Step(ScrX%,ScrY%),PelC(XRd%), BF
        Next XRd%
    Next YRd%
    Printer.EndDoc
```

ทำให้โปรแกรมสามารถทำงานได้เร็วกว่าการแสดงผลทีละจุด โดยคำสั่งที่ทำการแสดงผลจะถูกขั้นด้วยคำสั่งอื่น ๆ ที่ไม่ได้ทำงานประเภทเดียวกันภายในวนลูป ในการอ่านค่าสีจากแต่ละจุด (pixel) ของคอนโทรล picture box ก็จะใช้ฟังก์ชัน Point() และในการแสดงผลจุดที่อ่านค่าได้ออกทางเครื่องพิมพ์แทนที่ผู้เขียนจะใช้ประโยค Pset แต่ในที่นี้ได้ใช้ประโยค Line ทั้งนี้เนื่องจากต้องการให้ผู้อ่านได้เห็นถึงผลของค่าอัตราส่วนของแต่ละตำแหน่งแสดงผล (Twip per pixel) และสามารถนำค่าดังกล่าวมาใช้ได้อย่างถูกต้อง เพราะการพิมพ์อิมเมจด้วยวิธีนี้จะทำให้ได้อิมเมจที่มีอัตราส่วนของขนาดสมจริง เช่นเดียวกับที่ปรากฏภายในคอนโทรล picture box ส่วนวิธีที่จะพิมพ์อิมเมจให้ได้ขนาดเดียวกับที่



รูปที่ 4 แสดง Form1 เมื่อรูปทรง fractal ของ Sierpinski ถูกสร้างเสร็จแล้ว

ปรากฏในคอนโทรล picture box ผู้อ่านก็สามารถกระทำได้เช่นเดียวกัน โดยการเปรียบเทียบค่าของ Twip per pixel ที่ได้จากจอภาพ (โดยใช้ออปเจ็กต์ Screen) และที่ได้จากเครื่องพิมพ์ (โดยใช้ออปเจ็กต์ Printer) แต่อย่างไรก็ตาม การพิมพ์อิมเมจออกทางเครื่องพิมพ์โดยอาศัยวิธีในตัวอย่างที่ 8 ก็ยังไม่ใช่วิธีที่ดีและรวดเร็วที่สุด ทั้งนี้เนื่องจากวินโดวส์ได้เตรียงฟังก์ชันวินโดวส์ API ที่ช่วยในการจัดพิมพ์อิมเมจต่าง ๆ ออกทางเครื่องพิมพ์ ซึ่งผู้อ่านสามารถนำมาใช้กับ VB/Win ได้โดยไม่มีปัญหา วิธีนี้จะช่วยให้เกิดความยืดหยุ่นและเพิ่มความเร็วในการพิมพ์อิมเมจสูงมาก เพราะวินโดวส์สามารถจัดการกับงานดังกล่าวได้โดยตรง ซึ่งผู้เขียนก็จะได้นำมาอธิบายในตอนต่อ ๆ ไป

และถ้าหากผู้อ่านต้องการซ่อนคอนโทรล picture box ในขณะที่กำลังวาดรูปทรง fractal ก็ให้กดปุ่ม Hide สำหรับโปรแกรมตัวอย่างในตอนนี้ผู้อ่านจะเห็นว่าเทคนิคการซ่อนคอนโทรล picture box ก็ไม่ได้ช่วยให้การแสดงผลเร็วขึ้นสักเท่าไหร่เลย สุดท้ายถ้าหากผู้อ่านต้องการสิ้นสุดการทำงานของโปรแกรมก็กดปุ่ม Exit โปรแกรมก็จะจบการทำงาน ในรูปที่ 4 ได้แสดงรูปที่ได้จาก sierpinski gasket

ลิสติง SIER.MAK

```
SIER.BAS
ProjWinSize=127,1,248,126
ProjWinShow=0
```

ลิสติง SIER.BAS

```
Option Explicit
'
' Project Name   : SIER.MAK
' Program Name   : Sierpinski Fractal Monitoring
Program.
' File name      : SIER.FRM
' Written by     : SUTTHISAK PHONGTHANAPANICH
' CopyRight (C)1995 By SUTTHISAK
PHONGTHANAPANICH
'

' Global Constants.
Global Const DEFAULT = 0
Global Const HOURGLASS = 11
Global Const PIXELS = 3

Type COMPLEX
    a As Single
    b As Single
End Type
```

ลิสติง SIER.FRM

```
VERSION 2.00
Begin Form Form1
   AutoRedraw      =   -1  'True
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   3  'Fixed Double
   Caption         =   "Sierpinski Fractal
Generator"
   ControlBox      =   0   'False
   Height          =   4365
   Icon            =   0
   Left            =   1035
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   ScaleHeight     =   3960
   ScaleWidth      =   7530
   Top             =   1860
   Width           =   7650
   Begin CommandButton Command1
      Caption          =   "&Go"
      Default          =   -1  'True
      Height           =   555
      Left             =   4575
      TabIndex         =   17
      Top              =   2475
      Width            =   1395
   End
   Begin CommandButton Command2
      Caption          =   "&Hide"
      Height           =   555
      Left             =   4575
      TabIndex         =   15
      Top              =   3165
      Width            =   1395
   End
   Begin CommandButton Command4
      Caption          =   "E&xit"
      Height           =   555
      Left             =   6060
      TabIndex         =   14
      Top              =   3165
      Width            =   1395
   End
   Begin CommandButton Command3
      Caption          =   "&Print"
      Height           =   555
      Left             =   6060
      TabIndex         =   13
      Top              =   2475
      Width            =   1395
   End
   Begin Frame Frame1
      BackColor        =   &H00C0C0C0&
      Caption          =   " Coordinate
Initialization "
      ClipControls     =   0   'False
      Height           =   2265
      Left             =   4545
      TabIndex         =   1
      Top              =   0
      Width            =   2910
      Begin TextBox Text5
         Height           =   315
         Left             =   1950
         TabIndex         =   11
         Text             =   "Text5"
         Top              =   1575
         Width            =   890
      End
      Begin TextBox Text4
         Height           =   315
         Left             =   1950
         TabIndex         =   9
         Text             =   "Text4"
         Top              =   1245
         Width            =   890
      End
      Begin TextBox Text3
         Height           =   315
         Left             =   1950
         TabIndex         =   7
         Text             =   "Text3"
         Top              =   915
         Width            =   890
      End
      Begin TextBox Text2
         Height           =   315
         Left             =   1950
         TabIndex         =   5
         Text             =   "Text2"
         Top              =   585
         Width            =   890
      End
      Begin TextBox Text1
         Height           =   315
         Left             =   1950
         TabIndex         =   3
         Text             =   "Text1"
         Top              =   255
         Width            =   890
      End
      Begin Label Label7
         BackColor        =   &H00808080&
         BorderStyle      =   1  'Fixed Single
         Caption          =   "Label7"
         FontBold         =   0   'False
         FontItalic       =   0   'False
         FontName         =   "MS Sans Serif"
         FontSize         =   7.5
         FontStrikethru   =   0   'False
         FontUnderline    =   0   'False
         ForeColor        =   &H00FFFFFF&
         Height           =   300
         Left             =   45
         TabIndex         =   16
         Top              =   1905
         Width            =   2805
      End
      Begin Label Label5
         AutoSize         =   -1  'True
         BackStyle        =   0  'Transparent
         Caption          =   "Iteration"
         FontBold         =   0   'False
         FontItalic       =   0   'False
         FontName         =   "MS Sans Serif"
         FontSize         =   7.5
         FontStrikethru   =   0   'False
         FontUnderline    =   0   'False
         Height           =   210
         Left             =   1305
         TabIndex         =   10
         Top              =   1590
         Width            =   570
      End

      Begin Label Label4
         AutoSize         =   -1  'True
         BackStyle        =   0  'Transparent
         Caption          =   "Lower Right Y
(Imaginary)"
         FontBold         =   0   'False
         FontItalic       =   0   'False
         FontName         =   "MS Sans Serif"
         FontSize         =   7.5
         FontStrikethru   =   0   'False
         FontUnderline    =   0   'False
         Height           =   210
         Left             =   60
         TabIndex         =   8
```

ลิสติง SIER.FRM (ต่อ)

```
      Top             =   960
      Width           =   1815
   End
   Begin Label Label3
      AutoSize        =   -1  'True
      BackStyle       =   0   'Transparent
      Caption         =   "Lower Right X
(Real)"
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   7.5
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   210
      Left            =   405
      TabIndex        =   6
      Top             =   1275
      Width           =   1470
   End
   Begin Label Label2
      AutoSize        =   -1  'True
      BackStyle       =   0   'Transparent
      Caption         =   "Upper Left Y
(Imaginary)"
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   7.5
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   210
      Left            =   165
      TabIndex        =   4
      Top             =   645
      Width           =   1710
   End
   Begin Label Label1
      AutoSize        =   -1  'True
      BackStyle       =   0   'Transparent
      Caption         =   "Upper Left X
(Real)"
      FontBold        =   0   'False
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   7.5
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   210
      Left            =   510
      TabIndex        =   2
      Top             =   330
      Width           =   1365
   End
End
Begin PictureBox Picture1
   AutoRedraw      =   -1  'True
   BackColor       =   &H00FFFFFF&
   Height          =   3645
   Left            =   0
   ScaleHeight     =   241
   ScaleMode       =   3   'Pixel
   ScaleWidth      =   298
   TabIndex        =   0
   Top             =   270
   Width           =   4500
End
Begin Label Label6
   Alignment       =   2   'Center
   BackColor       =   &H00FF0000&
   Caption         =   "Sierpinski Fractal
Monitor"
   ForeColor       =   &H00C0C0C0&
   Height          =   270
   Left            =   0
   TabIndex        =   12
   Top             =   0
   Width           =   4500
End
End
Option Explicit
'
' Project Name   : SIER.MAK
' Program Name   : Sierpinski Fractal Monitoring
Program.
' File name      : SIER.FRM
' Written by     : SUTTHISAK PHONGTHANAPANICH
' CopyRight (C)1995 By SUTTHISAK PHONGTHANAPANICH
'

Dim TerminatedFlag%

Sub Command1_Click ()

    If Command1.Caption = "&Go" Then
        TerminatedFlag% = False
        Command1.Caption = "&Cancel"
        CreatSierpinski
    Else
        TerminatedFlag% = True
        Command1.Caption = "&Go"
    End If

End Sub

Sub Command2_Click ()

    If Command2.Caption = "&Hide" Then
        Command2.Caption = "&Show"
        Picture1.Visible = False
    Else
        Command2.Caption = "&Hide"
        Picture1.Visible = True
    End If

End Sub

Sub Command3_Click ()

    ' Declare local variables.
    Dim XRd%, YRd%, ScrX%, ScrY%
    Dim PelX%, PelY%, PixX%, PixY%

    Screen.MousePointer = HOURGLASS
    ' Find numbre of pixel in picture control.
    PixX% = Picture1.ScaleWidth - 1
    PixY% = Picture1.ScaleHeight - 1
    ' Declare and initialize screen resolution
variables.
    ScrX% = Screen.TwipsPerPixelX
    ScrY% = Screen.TwipsPerPixelY
    ' For each row of pixels in the source
bitmap.
    For YRd% = 0 To PixY%
        If YRd% Mod 2 Then
            Label7.Caption = "Status
[Printing...]"
        Else
            Label7.Caption = "Status [Printing
...]"
        End If
        Label7.Refresh
        PelY% = Picture1.Top + (YRd% * ScrY%)
        ' For each pixel in the current row of
the source bitmap.
        ReDim PelC(0 To PixX%) As Long
        For XRd% = 0 To PixX%
```

⇨

ลิสติง SIER.FRM

```
            ' Store the pixel color in a local
variable.
            PelC(XRd%) = Picture1.Point(XRd%,
YRd%)
        Next XRd%
        For XRd% = 0 To PixX%
            PelX% = Picture1.Left + (XRd% *
ScrX%)
            Printer.Line (PelX%, PelY%)-
Step(ScrX%, ScrY%), PelC(XRd%), BF
        Next XRd%
    Next YRd%
    Printer.EndDoc
    Label7.Caption = "Status [Ready]"
    Screen.MousePointer = DEFAULT

End Sub

Sub Command4_Click ()

    Unload Me

End Sub

Function ComplexFormat$ (Za As Single, Zb As
Single)
    '
    Dim msg$

    msg$ = Format$(Za, "#0.0000") & " + "
    msg$ = msg$ & Format$(Zb, "#0.0000") & "i"

    ComplexFormat$ = msg$

End Function

'
' Generates Sierpinski set according to the
parameters entered by the user.
'
Sub CreatSierpinski ()

    Dim PixX%, PixY%, X%, Y%, i%
    Dim IterNum%, InSetFlag%
    Dim RealRange As Single, ImagRange As Single
    Dim DeltaR As Single, DeltaI As Single
    Dim Z As COMPLEX, Z1 As COMPLEX, Zn As
COMPLEX

    Picture1.Cls

    ' Find numbre of pixel in picture control.
    PixX% = Picture1.ScaleWidth - 1
    PixY% = Picture1.ScaleHeight - 1

    ' Calculate width and height of coordinate.
    RealRange = Abs(Val(Text3.Text) -
Val(Text1.Text))
    ImagRange = Abs(Val(Text4.Text) -
Val(Text2.Text))
    DeltaR = RealRange / (PixX% + 1)
    DeltaI = ImagRange / (PixY% + 1)

    IterNum% = Val(Text5.Text)
    Z.a = Val(Text1.Text)
    ' Begin the main processing loop.
    For X% = 0 To PixX%
        If X% Mod 2 Then
            Label7.Caption = "Status
[Processing...]"
        Else
            Label7.Caption = "Status [Processing
...]"
        End If
        Z.a = Z.a + DeltaR
        Z.b = Val(Text2.Text)
        ReDim ColorArr(0 To PixY%) As Integer
        For Y% = 0 To PixY%
            DoEvents
            If TerminatedFlag% Then GoTo
SubTerminated
            Z.b = Z.b - DeltaI
            Z1 = Z
            InSetFlag% = True
            For i% = 1 To IterNum%
                If Z1.b > .5 Then
                    Zn.a = 2 * Z1.a
                    Zn.b = 2 * Z1.b - 1
                ElseIf Z1.a > .5 Then
                    Zn.a = 2 * Z1.a - 1
                    Zn.b = 2 * Z1.b
                Else
                    Zn.a = 2 * Z1.a
                    Zn.b = 2 * Z1.b
                End If
                If ModulusComplex(Zn) > 2 Then
                    InSetFlag% = False
                    Exit For
                End If
                Z1 = Zn
            Next i%
            ' save pixel color.
            If InSetFlag% Then ColorArr(Y%) = 0
Else ColorArr(Y%) = (i% Mod 15) + 1
        Next Y%
        ' show each column.
        For Y% = 0 To PixY%
            Picture1.PSet (X%, Y%),
QBColor(ColorArr(Y%))
        Next Y%
        ' Refresh the picture.
        Picture1.Refresh
    Next X%

SubTerminated:
    If Command1.Caption <> "&Go" Then
Command1.Value = True
    Label7.Caption = "Status [Ready]"

End Sub

Sub Form_Load ()

    Left = (Screen.Width - Width) \ 2
    Top = (Screen.Height - Height) \ 2

    Text1.Text = "-0.15216"
    Text2.Text = "1.1216"
    Text3.Text = "1.10216"
    Text4.Text = "-0.1216"
    Text5.Text = "35"
    Label7.Caption = "Status [Ready]"

End Sub

Sub Form_Unload (Cancel As Integer)

    End

End Sub

Function ModulusComplex (X As COMPLEX) As Single

    ModulusComplex = Sqr(X.a * X.a + X.b * X.b)

End Function
```